Samuel Ogunfunmi

Purpose:

Date: 11/10/20

**Problem 1)**

**1. Algorithm (Solution Plan for the Problem):**

1. Ask user for input
2. Create function, running total variable, and dictionary
3. Use for loop to add to running total and identify each letter and its score for the input
4. Send total back to main
5. Print total

**2. Program Source Code (copy and paste from IDE):**

def scrabble(word):

total = 0

word = word.upper() #makes input upper case to match dictionary

score = {'A': 1, "E": 1, "I": 1, "L": 1, "N": 1, "O": 1, "R": 1,

"S": 1, "T": 1, "U": 1, "D": 2, "G": 2,"B": 3, "C": 3,

"M": 3, "P": 3,"F": 4, "H": 4,"V": 4, "W": 4, "Y": 4,

"K": 5,"J": 8, "X": 8, "Q": 10, "Z": 10}

for i in word:

score[i]

total += score[i]

return total

def main():

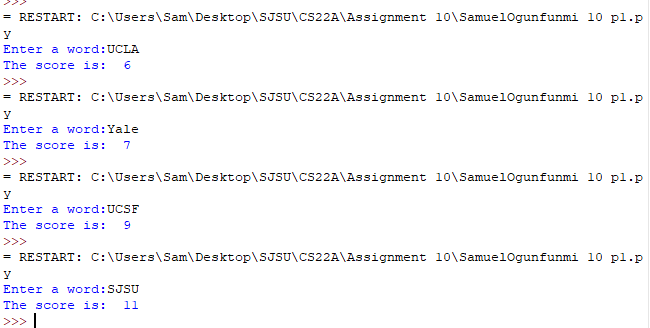
word = input("Enter a word:")

scrabblescore = scrabble(word)

print("The score is: ",scrabble(word))

main()

**3. Program Output Screenshots/Screen Print(s) and/or Error Messages:**



**Problem 2)**

**1. Algorithm (Solution Plan for the Problem):**

1. Ask user for a number
2. Create a dictionary
3. Use for loop to generate and displays a dictionary that contains keys (from 1 and n) and values in the form n\*n

**2. Program Source Code (copy and paste from IDE):**

def main():

n = int(input("Enter a number: "))

a = dict()

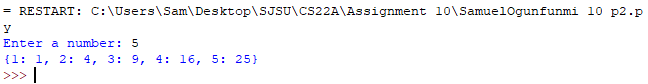
for x in range(1,n+1):

a[x] = x \* x

print(a)

main()

**3. Program Output Screenshots/Screen Print(s) and/or Error Messages:**



**Problem 3)**

**1. Algorithm (Solution Plan for the Problem):**

1. Create a nested dictionary for songs
2. Ask the user for a song
3. Use nested for loop and if statements to find if user input is found in the dictionary
4. If found display the song name and artist
5. If not found display that the song is not found in the music library

**2. Program Source Code (copy and paste from IDE):**

def main():

SongLibrary = {'Drake': {'Headlines','Emotionless','Do Not Disturb',},

'Smino':{'Wild Irish Roses','Anita','Father Son Holy Smoke'},

'Bryson Tiller':{'Dont','Things Change','Sorry Not Sorry',},

'Blxst':{'Overrated','Hurt','No Love Lost',},}

print(SongLibrary)

for i in range(3):

song = input("Enter a song: ")

for artist, songs in SongLibrary.items():

if song in songs:

print(f'Found "{song}" under artist {artist}.')

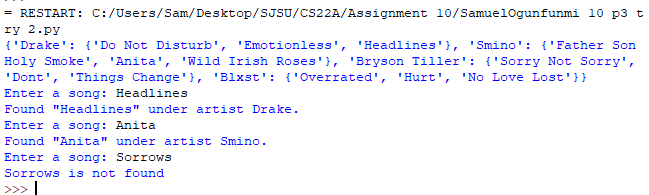
break

if song not in songs:

print(song, "is not found")

main()

**3. Program Output Screenshots/Screen Print(s) and/or Error Messages:**

****

**Problem 4)**

**1. Algorithm (Solution Plan for the Problem):**

1. Create a nested dictionary

**2. Program Source Code (copy and paste from IDE):**

Counties = {{'Alameda': {'Livemore':{'90,269'},'Dublin':{'63,445'},'Pleasanton':'{82,372}'},},

{'Santa Clara':{'Santa Clara':{'129,488'},'San Jose':{'1,030,000'}, 'Sunnyvale':{'153,185'},},},

{'Sonoma':{'Sonoma':{'11,248'},'Santa Rosa':{'177,586'},'Petaluma':{'61,917'},},},}

**3. Program Output Screenshots/Screen Print(s) and/or Error Messages:**

**Problem 5)**

**1. Algorithm (Solution Plan for the Problem):**

1. Ask user for input
2. Create function
3. Create dictionary in function
4. Use for loop identify how many times a letter is used in string
5. Send total to main
6. Print total

**2. Program Source Code (copy and paste from IDE):**

def phrasecounter(string):

a = {}

for i in string:

a[i] = a.get(i, 0) +1

return a

def main():

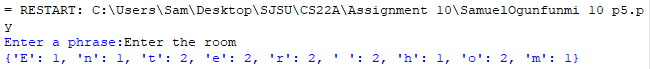
string = input("Enter a phrase:")

phrasecounter(string)

print(phrasecounter(string))

main()

**3. Program Output Screenshots/Screen Print(s) and/or Error Messages:**

****

**Conclusion/What you learned writing this program and what problems you encountered.**